FIFTY YEARS OF SERVICE
1940 - 1990

Research Product 90-07

User's Manual for the Prototype Analyst Workstation (PAWS)

February 1990

Fort Huachuca Field Unit
Systems Research Laboratory

U.S. Army Research Institute for the Behavioral and Social Sciences

Approved for public release; distribution is unlimited

U.S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES

A Field Operating Agency Under the Jurisdiction of the Deputy Chief of Staff for Personnel

EDGAR M. JOHNSON **Technical Director**	**JON W. BLADES** **COL, IN** **Commanding**

Research accomplished under contract for
the Department of the Army

Vector Research, Inc.

Technical review by

Arthur Marcus
Michael L. Mather

NOTICES

~~**DISTRIBUTION**: Primary distribution of this report has been made by ARI. Please address correspondence concerning distribution of reports to: U.S. Army Research Institute for the Behavioral and Social Sciences, ATTN: PERI-POX, 5001 Eisenhower Ave., Alexandria, Virginia 22333-5600.~~

FINAL DISPOSITION: This report may be destroyed when it is no longer needed. Please do not return it to the U.S. Army Research Institute for the Behavioral and Social Sciences.

NOTE: The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION
Unclassified

1b. RESTRICTIVE MARKINGS
--

2a. SECURITY CLASSIFICATION AUTHORITY
--

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE
--

3. DISTRIBUTION / AVAILABILITY OF REPORT
Approved for public release;
distribution is unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
VRI-ARI-6 FR82-1

5. MONITORING ORGANIZATION REPORT NUMBER(S)
ARI Research Product 90-07

6a. NAME OF PERFORMING ORGANIZATION
Vector Research, Inc.

6b. OFFICE SYMBOL *(If applicable)*
--

7a. NAME OF MONITORING ORGANIZATION
U.S. Army Research Institute
Field Unit at Fort Huachuca

6c. ADDRESS *(City, State, and ZIP Code)*
P.O. Box 1506
Ann Arbor, MI 48106

7b. ADDRESS *(City, State, and ZIP Code)*
ATTN: PERI-SA
Ft. Huachuca, AZ 85613-7000

8a. NAME OF FUNDING / SPONSORING ORGANIZATION
U.S. Army Research Institute for the Behavioral and Social Sciences

8b. OFFICE SYMBOL *(If applicable)*
PERI-S

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
MDA903-81-C-0579

8c. ADDRESS *(City, State, and ZIP Code)*
5001 Eisenhower Avenue
Alexandria, VA 22333-5600

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
63739A	793	143	C1

11. TITLE *(Include Security Classification)*
User's Manual for the Prototype Analyst Workstation (PAWS)

12. PERSONAL AUTHOR(S)
Thompson, Dave; Rueter, Henry; Rainaldi, William; and Orosz, Joseph (Vector Research, Inc.)

13a. TYPE OF REPORT
Final

13b. TIME COVERED
FROM 84/03 TO 85/09

14. DATE OF REPORT *(Year, Month, Day)*
1990, February

15. PAGE COUNT

16. SUPPLEMENTARY NOTATION
This project was technically monitored by Beverly Knapp and Jo Hall

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Situation assessment
Intelligence analysis
Decision aid
Multi-attribute decision theory

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

The Prototype Analyst Workstation (PAWS) is a personal-computer-based decision aid intelligence analysts can use to evaluate alternative enemy courses of action. It assists analysts in assigning weights to attributes and scoring alternative enemy courses of action so that they can compute an overall evaluation of each course of action. PAWS also has a facility for Bayesian updating of probabilities over the courses of action as the analyst receives new intelligence information. This report is a manual for analysts who need to understand how to run PAWS to solve intelligence problems, computer system managers who need to understand how to configure PAWS and its associated files on a computer system, and for persons who need to develop new materials for use with PAWS (i.e., new text and graphics descriptions of problem scenarios and new hierarchies of factors for evaluating enemy courses of action).

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Beverly G. Knapp

22b. TELEPHONE *(Include Area Code)*
(602) 538-4704

22c. OFFICE SYMBOL
PERI-SA

Previous editions are obsolete.

Research Product 90-07

User's Manual for the Prototype Analyst Workstation (PAWS)

Dave Thompson, Henry Rueter, William Rainaldi,
and Joseph Orosz
Vector Research, Inc.

Field Unit at Fort Huachuca, Arizona,
Julie A. Hopson, Chief

Systems Research Laboratory
Robin L. Keesee, Director

U.S. Army Research Institute for the Behavioral and Social Sciences
5001 Eisenhower Avenue, Alexandria, Virginia 22333-5600

Office, Deputy Chief of Staff for Personnel
Department of the Army

February 1990

Army Project Number
2Q263739A793

Human Factors in Training and
Operational Effectiveness

FOREWORD

The Army Military Intelligence (MI) community is introducing a substantial number of ADP-based Intelligence/Electronic Warfare (IEW) systems for the 1990s to assist in the collection and processing of enemy intelligence. An important aspect of automation support for intelligence operations is development of workstations for analysis of collected data.

The U.S. Army Research Institute for the Behavioral and Social Sciences (ARI) Field Unit at Fort Huachuca research development effort provides support to the intelligence community by conducting projects that identify human performance requirements for emerging IEW systems and evaluate the impact of new IEW systems and interface architectures on MI soldiers. One important aspect of this work program concerns the enhancement of IEW workstation interfaces and methods to improve analysis capability. This is identified in the research Letter of Agreement between ARI and USAICS formulated in 1984.

The development of the ARI multi-attribute utility-based workstation concept for intelligence analysts (PAWS) was driven by numerous requests in 1984-86 from U.S. Army Intelligence Center and School (USAICS) combat developers. The work was sponsored by the USAICS Artificial Intelligence (AI) group, which is concerned with development of modules suitable for situation development functions performed by intelligence analysts. The initial versions of the workstation were briefed to the Commanding General (CG), USAICS in FY 85, who recommended its inclusion in the AI situation development modules being developed by USAICS for IEW systems.

Continuing ARI work in the MI area will include support to USAICS in evaluation of these and other analyst workstation capabilities.

Edgar M Johnson

EDGAR M. JOHNSON
Technical Director

USER'S MANUAL FOR THE PROTOTYPE ANALYST WORKSTATION (PAWS)

CONTENTS

LIST OF EXHIBITS

USER'S MANUAL FOR THE PROTOTYPE ANALYST WORKSTATION (PAWS)

PART I

INTRODUCTION

1.0 INTRODUCTION

The Prototype Analyst Workstation (PAWS) is a computer program that runs on the IBM PC XT computer. Employing multi-attribute utility analysis and a sophisticated information display system, it can be used to help an intelligence analyst assess battlefield situations and intelligence, teach students the analysis process, or investigate the cognitive requirements of the assessment task.

1.1 OVERVIEW

This guide for users of the Prototype Analyst workstation (PAWS) was written with two different kinds of PAWS users in mind. The first kind of user, the system manager, will be concerned with configuring or modifying the PAWS information display for later use by the analyst. Thus, the system manager will be concerned witn such things as developing new models (Part II), changing the menu structure for displaying scenario information (Part III), or creating new graphics for use with PAWS (Part IV).

On the other hand, the second kind of PAWS user, the analyst (or student or psychological subject), will simply want to use the program as a problem-solving aid. Thus, the analyst will want to read about such things as how to use PAWS to analyze a given problem (Part II), or how to use the PAWS augmented information display (Part III).

This manual is divided into five parts. Following this introduction, the next two parts cover the needs of both system manager and analyst; Part II, "A User's Manual to PAWS," describes both the theory that underlies PAWS as well as the practical details of running PAWS on the

IBM PC XT computer. Part III, "A Guide to Augmented Information Display on PAWS," tells both how to use and how to configure the Augmented Information Display feature of the program.

The last two parts of the manual are for the system manager. Part IV, "TREEDEF: Tree Utility for Defining and Editing Problem Structures," shows how to create, edit, store, and delete the tree structures that form the basis for PAWS' multi-attribute utility analysis. The last part, Part V, "A User's Manual for the ARTIST Program," is a detailed description of an extensive, independent program that makes it easy to create new graphics displays for PAWS.

1.2 THE PROGRAM INTERFACE CONTROLLER

The Program Interface Controller (PIC) can be used to control a user's access to programs resident on the current floppy and/or hard disks. It does this by presenting a menu of functions that the user can perform, reading his response, then setting up a batch file to perform the DOS commands necessary to perform the chosen function. PIC is especially suited to controlling the analyst's access to a number of different PAWS scenarios. This section is concerned with the steps that the system manager needs to take to configure PIC for this and other uses.

The major step that the systems operator will need to perform is the creation of the file that tells PIC how to create both the batch files and the menu. This file has to reside on the same disk as the files 'PIC.BAT' and 'PIC- PROG.COM' and must be named 'MENUFILE.TXT.' The structure of this file will be discussed later. The distribution disk is set up so that the program will run when the user types 'PIC.'

The system can be made to run when the user turns on the computer by simply adding the line

```
PIC
```

to the end of the boot disk's AUTOEXEC.BAT file, and putting the files 'PIC.BAT', 'PIC-PROG.COM', and the 'MENUFILE.TXT' that you have created onto the boot disk. If there is no AUTOEXEC.BAT file on the boot disk, create one containing the line above. The boot disk must also have some space remaining on it, usually about 4K will suffice. If you set up some of the menu responses to issue a complicated series of DOS commands, you may find that more than 4K will be necessary.

The structure of MENUFILE.TXT is quite simple. It is a text file that can be created using any available text editor. It must consist of a series of lines to be printed on the menu, each followed by the batch commands to be performed when that menu choice is selected. Each of the menu lines must start with the character '$', and all lines until the next menu line will be interpreted by the program as batch commands. That means that if any line which is not preceded by a '$' is also not a batch command, the user will see extraneous error messages when DOS attempts to interpret these lines as batch commands. This is because the program does not check the lines for validity as DOS commands. You must debug your MENUFILE.TXT files to make sure that they work properly.

The following is an example MENUFILE.TXT that serves as an interface between three separate programs. NOTE: the numbers should not appear in the actual file, they are only for reference.

```
(1)   $Run PAWS - Jayhawk scenario.
(2)   CHDIR JAYHAWK
(3)   PAWS
```

```
(4)   CHDIR ?

(5)   $Run PAWS - IRAN?IRAQ scenario.

(6)   CHDIR IRAN-IRA

(7)   PAWS

(8)   CHDIR ?

(9)   $Run AIENCOA.

(10)  CHDIR AIENCOA

(11)  AIENCOA

(12)  CHDIR ?
```

This file would produce a menu that looks like the following:

```
(0) Exit the Controller to the system.

(1) Run PAWS - Jayhawk scenario.

(2) Run PAWS - IRAN?IRAQ scenario.

(3) Run AIENCOA.
```

The sample MENUFILE.TXT shifts control between three different sub-directories on a disk (in this case a hard disk), two of which contain copies of PAWS with different scenario and TGD files. This is controlled by lines 2-4 and 6-8. The third choice on the menu will send the user to a third sub-directory which contains a program called AIENCOA. This is controlled by lines 10-12. The menu is created from lines 1, 5, and 9. Note that in each case the batch files return to the root directory, which contains PIC.BAT (lines 4, 8, and 12). This must be done if the program is to work correctly. When the batch file is done you must be back to the disk and sub-directory that you were in when the batch file started.

One final item is that PIC creates a temporary batch file called MENUTEMP.BAT. Once the program is finished this file is no longer needed and can be deleted if you wish. However you must not try to delete this file while the program is running (i.e. from MENUFILE.TXT). This could cause serious problems, such as locking up the system, which would necessitate rebooting.

PART II

A USER'S MANUAL TO PAWS

1.0 INTRODUCTION

The Prototype Analyst Workstation (PAWS) is a computer program for the IBM PC XT computer designed to assist intelligence analysts in making battlefield situation assessments. PAWS assists the analyst by prompting him to define the alternative courses of action open to the enemy, after which it leads the analyst through a procedure that results in an estimate of these probablilites of these courses of action. There are two stages in the estimation of these probabilities: (1) examining the courses of action from the enemy commander's point of view leads to an initial model of the situation; and (2) assessing the impact of succeeding items of new intelligence produces updated probabilites of the courses of action.

The remainder of this part contains a description of the PAWS methodology (section 2.0); instructions for the use of PAWS (section 3.0); and a guide for programmer-analysts who may need to know how to set up PAWS to run on the XT computer (section 4.0).

2.0 THE PAWS APPROACH

PAWS provides a two-phased approach to the evaluation of alternative courses of action: first, an "initial model" is built from the information initially available; and, second, the initial model is updated to assess the impact of new intelligence. A new model, specifying the probabilities of the alternative courses of action, is created after assessing the impact of each successive item of new intelligence. The following discussion outlines the methods used in creating the initial model (subsection 2.1) and assessing the impact of new intelligence (subsection 2.2).

2.1 THE INITIAL MODEL

When you begin a PAWS analysis, you choose one of a list of available problem structures to guide your work. For IPB analysis it will often be useful to choose the ENCOA Tree, while for other problems other structures may be used. For example, the ENCOA (Enemy Courses of Action) structure contains 25 factors, organized into five categories, on which you are to score each enemy course of action.

TERRAIN FACTORS
- Fields of fire
- Cover and concealment
- Mobility
- Seize/deny key terrain
- Observation
- Natural/artificial obstacles

US FORCE FACTORS
- Disposition
- Strength and condition
- Reserves
- Logistic support
- Probable actions/reactions
- Command/control

OPFOR FACTORS
Current disposition
Strength and condition
Reserves
Logistic support
Command/control

WEATHER FACTORS
Observation/visibility
Cover and concealment
Mobility
Extreme weather conditions

RISK FACTORS
US actions/reactions
Dependence on other commands
Dependence on surprise/deception
Unexpected weather

Other sets of evaluation factors are available in PAWS, and still more can be created with the help of a separate program (Part IV of this manual).[1]

Creating the initial model requires two steps: (1) scoring the COAs; and (2) assigning weights to the categories and factors. Scoring considers each of the (bottom-level) factors, one at a time, and scores all the COAs according to their relative importance for the COA, on a scale of zero to 100. Weighting considers each of the categories, one at a time, and weights the factors subordinate to that category according to their relative importance in contributing to that category. All the categories are then weighted according to their relative importance to the overall evaluation.

A factor's total contribution to the overall evaluation is the product of its weights within its category and of its category's weight will respect to the total problem. The relative importance of each COA is the

[1] The other program is TREEDEF, for Tree Definition Program. The "Tree" terminology refers to the tree-like structure of the hierarchy of evaluation factors.

sum of its weighted scores across all the evaluation factors. This relative importance is output by the program as the probability of the course of action.

2.2 ASSESSING THE IMPACT OF NEW INTELLIGENCE

After creation of the initial model, you can use PAWS to update the model to assess the impact of newly received intelligence relating to the courses of action. The updating procedure prompts you to estimate how much more likely the receiving the new intelligence is for the different COAs, and it uses your estimates to update the probabilities of the COAs. The procedure involves the following steps:

(1) The program shows you a list of the probabilities of the COAs initial model and subsequent updates.

(2) You are asked to rank the COAs according to how likely you think the receipt of the new intelligence was, if each COA were the one being followed.

(3) The program asks you to estimate how many times more likely the new intelligence was, under the COA with the highest probability of the intelligence occurring, compared to the COAs with the second, third, etc. highest probabilities of the new intelligence occurring.

(4) The program calculates the updated COA probabilities, displays them, and asks if you are satisfied with them. If you are satisfied, the program stores the results for reference on future sessions, and the update is completed.

(5) If you respond that you are not satisfied, the program gives you a chance to correct the answer. It gives you a choice of either (a) starting over at step two, above, or (b) overriding this procedure by specifying the COA probabilities directly.

3.0 OPERATION OF THE PROGRAM

In most cases the use of PAWS follows two broad phases: building an initial model of the situation, followed by assessing the impact of new intelligence received following the creation of the initial model. The process begins when a user (say, an intelligence analyst) gathers the information at his disposal and uses PAWS to refine the information into an initial model of the battlefield situation. He then reviews the results of the assessment and, if not satisfied with any of the details, he edits the model to correct the unsatisfactory parts. He then tells the program to save the assessment -- "The initial model" -- on the built-in "hard" disk of the XT computer for future reference. Nine different models can be stored; old models can be erased to make room for new ones.

The next broad phase, assessing the impact of new intelligence, can be taken immediately or on a later computer session (by loading the saved initial model into the program). This phase updates the initial model to take account of new intelligence received since the creation of the initial model, and it can be repeated until all items of new intelligence have been processed. After each item of new intelligence is assessed, the program automatically saves the updated model, along with a history of preceding updates, so that the procedure can be resumed at the same point on later sessions at the computer.

The remainder of this section expands on this summary of program operation.

3.1 THE MAIN MENU

The program is controlled from a MAIN MENU, offering the following options:

(1) Introduction
(Lets you see an introduction.)

(2) Create Model
(This is how you start a new problem.)

(3) Edit Selected Items
(Before you save the model for later use, you can review it and make changes:
(1) Review scores
(2) Edit scores
(3) Edit weights
(4) Edit COA labels
(5) Delete COA)

(4) Save Model
(Stores your model for later use.)

(5) Load Model
(Retrieves a saved model.)

(6) Assess Impact of New Intelligence
(After you have saved your initial model, you can update the COA probabilities to reflect new intelligence.)

(7) Display Results
(Shows you the results of the model you have created:
(1) Display initial model information
(2) Rank by relative importance
(3) Rank by selected COAS
(4) Perform sensitivity analysis
(5) Impact of new Intelligence)

(8) Print Results
(Prints what Display Results shows on the screen.)

(9) Erase Model
(Permanently erases a save model.)

(0) Exit Program
(Ends the session.)

3.2 TO DEVELOP A NEW MODEL

To start the analysis of a new problem, choose the CREATE MODEL option. The program will prompt you to go through the following steps:

(1) Select a problem structure (i.e., a hierarchy of factors and categories, such as the ENCOA structure) from a list of structures available; (if a prearranged problem has been set up with text and graphics displays to explain the problem, it will be item zero, "The Sided Problem," on the list.)

(2) Specify the number of COAs to be considered;

(3) Give long and short names to the COAs;

(4) Score each COA on the factors of the evaluation structure that you selected in the first step; and

(5) Weigh the importance of these factors and the categories within which they are grouped.

If a prearranged problem has been set up on the computer with explanatory text and graphics displays (which you can view while developing the model), it will appear on the list of problem structures as item zero, "The Aided Problem." If you choose to work on The Aided Problem, you do not have to complete steps two and three, since the number of COAs and their names will have been set up for you ahead of time. In this case the program simply tells you the names of the COAs, along with the name of a scenario which they apply. In any case, you must go through steps four and five (scoring and weighting) to create the model.

Next, you can DISPLAY RESULTS of the initial model. The probabilities for the enemy courses of action are calculated from your inputs. These COA evaluations represent your subjective estimate of the enemy situation.

Also available in the DISPLAY RESULTS option are three types of feedback about your inputs. One is a list of the factors rank ordered in terms of their contribution to the COA evaluations. Second is a list of the factors rank ordered in terms of the extent to which they distinguish two specified COAs. Third is a sensitivity analysis that indicates how stable your judgments are, alerting you to those factors for which a small shift in importance could result in a significant change in COA probabilites. The feedback options (factors ranked by importance, factors ranked by selected COAs, and sensitivity analysis) pertain only to the initial model developed before assessing the impact of new intelligence.

If you are not satisfied with the initial model, you can choose the EDIT SELECTED ITEMS from the main menu and alter the weights or scores you have supplied.

Once you have examined the implications of your inputs using the aid's feedback options, then SAVE THE MODEL you created.

Now you are ready to ASSESS THE IMPACT OF NEW INTELLIGENCE on the COA probabilites as new data is received. You can enter up to 20 data messages per model. The aid will automatically save them.

When you have completed your analysis, you can DISPLAY RESULTS of the updated model. The aid will provide you with a history of the COA probabilities.

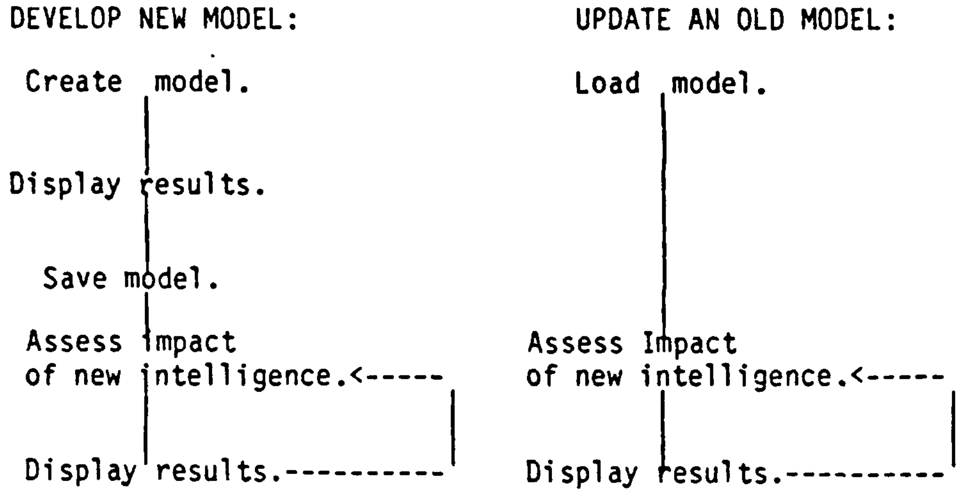
3.3 TO UPDATE AN OLD MODEL

Choose the Load Model option and load the model you want into the computer's working memory.

Then, ASSESS THE IMPACT OF NEW INTELLIGENCE on the COA probabilities. You can continue until you have entered a total of 20 messages.

When you are finished, DISPLAY THE RESULTS of the updated model.

The following diagrams compare the procedures for developing a new (initial) model and updating an old model:



DEVELOP NEW MODEL:

Create model.

Display results.

Save model.

Assess impact of new intelligence.<-----

Display results.----------

UPDATE AN OLD MODEL:

Load model.

Assess Impact of new intelligence.<-----

Display results.----------

3.4 TEXT AND GRAPHICS AID

When you choose the CREATE MODEL option the program prompts you to select one of a list of problem structures for scoring the COAs. If you choose number zero, "The Aided Problem," you will be led through a prearranged problem.

If you choose the aided problem, you will not have to specify the problem structure, the number of COAs, or their names. These things will have been set up for you.

You can view text and graphics explaining the aided problem either (a) when you are shown an introductory menu of scenario materials, upon starting to create the model; or (b) when you enter H (for Help) at certain times while creating and using the model.

You can enter H and see text and graphics scenario help when you are scoring the COAs and when you ASSESS THE IMPACT OF NEW INTELLIGENCE.

3.5 USE OF THE COMPUTER

Procedures for running the program depend in part on the way it has been installed on the XT computer. Nevertheless, once the computer has been turned on, there should be little more involved than entering the name of the program through the keyboard of the computer. That is, type the word "PAWS" and press the RETURN key.

After a delay of a few seconds, while the program is being initialized for use, the main menu of the program will appear on the screen.

Whenever the program asks you for a response, you must press the RETURN key after typing the response. If you make an error, you can backspace and correct the entry before pressing the RETURN key. If you are unfamiliar with the keyboard of the XT, ask someone to point out the RETURN and BACKSPACE keys to you, as the symbols on them may not be familiar.

If there is a printer attached to your computer, you can obtain a printed copy of the output tables produced by the PRINT RESULTS item (item eight) of the main menu. However, should you need a copy of the entire session -- both your typed entries and the program's output to the screen -- or of any portion of the session, you can obtain a copy on the printer very simply. When you hold down the CTRL key and press the PRTSC key, everything that appears on the screen thereafter will also be printed. This process will continue until you enter the CTRL PRTSC combination again to turn off the printing. If you should want only the current contents of the screen to be printed, hold down the SHIFT key and press the PRTSC key. Only the current screen contents will be printed.

4.0 INSTRUCTIONS FOR SYSTEM MANAGERS

Paws was designed for use on IBM PC XT computers with at least 256 K RAM memory and an IBM color graphics monitor, or a similar monitor compatible with the IBM color graphics adapter. The XT computer has a "hard" disk drive, which is needed to run the program. (The PAWS program, plus various data files used by PAWS, exceeds the space available in two floppy diskette drives.) The program was developed to run under the DOS 2.0 operating system, and requires so much space that other programs cannot run conveniently.

The PAWS program, and the files which it uses, should all reside on the same subdirectory of the hard disk, which is drive C: of the XT computer. (If you have two hard disks on your machine, be sure to use drive C: for PAWS.) Because of the number of files involved, it is a good idea to set aside a subdirectory for PAWS alone. If you receive distribution diskettes for PAWS and need to install the program on your XT computer, simply copy the contents of all of the diskettes into the subdirectory you have set aside for PAWS. The program can then be run by entering the name "PAWS," which is the name of the executable code file.

The files in the PAWS system fall into a few basic categories, according to the three-letter extension that follows each file name.

FILE	CONTENTS
PAWS.EXE	Executable code file
MODELS.DAT	User-defined models
TREEFILE.DAT	"Tree" structures of factors and categories, from which the user chooses one when creating a model. Contents are changed by the TREEDEF utility.

xxx.BET

Several files containing texts and menus printed by PAWS.

PROBLEM. TGD
MESSAGES. TGD

Two files that are optional; if present, they define a set of text and graphics aids that can assist the user in applying PAWS to a particular scenerio.

xxx.TXT[1]

Text files referenced by the TGD files.

xxx.GPH[1]

Graphics files referenced by the TGD files.

USERSTAT.DAT

Chronological Journal of user interactions with the text and graphics display aid. File is emptied and started over by the program at each session.

[1] The TGD, TXT, and GPH files are optional. If no PROBLEM.TGD file is present, no display aid is available to the user, but unaided problems can be done.

5.0 THE USER JOURNAL FOR THE PROTOTYPE ANALYST WORKSTATION

5.1 INTRODUCTION

The Prototype Analyst Workstation (PAWS) can produce a user journal, a history of user activities during a PAWS session. The journal is a file containing a chronological list of the user actions, such as viewing a graphics or text aid to the scenario, that occur during the session. Production is automatic and is not evident to the user. Following a PAWS session, experimenters can examine the journal, use it as input to their statistical analyses, and print it or store it on diskette for future reference.

5.2 FORMAT OF THE JOURNAL

The journal begins with a header giving the date of the session, followed by a chronological list of user events, one line per event. Each event is identified by the following items:

(a) the time of day in hours/minutes/seconds;

(b) the name of the event ("Start viewing" or "End viewing");

(c) descriptors of the event (the name of the file viewed and the name of the text or graphics contained in the file).

The format of the journal can be seen on exhibit 1 below. One event is recorded when the user selects a text or graphic aid and it appears on the screen, and another event is recorded when he elects to stop viewing it. "Start viewing" and "End viewing" lines occur in the journal for these two kinds of events.

To identify the viewed material, the journal also provides the name of the file being viewed and the name of the file's contents. If the

EXHIBIT 1: FORMAT OF USER EVENT MESSAGES IN THE USER JOURNAL

Columns	Contents
1-8	Time (hh/mm/ss)
11-29	Type of event (start viewing, end viewing)
39-50	Name of text or graphics file
53-	Title of document in file

EXAMPLE USER JOURNAL

```
                    HISTORY OF USER EVENTS

                      DATE = 11-16-84

13:22:32  Start of session
13:22:33  Start viewing          File =   FULDA0.TXT   Introductory Text
13:22:35  End viewing            File =   FULDA0.TXT   Introductory Text
13:23:10  Start viewing          File =   FULDA3.GPH   Corps Deployments
13:23:25  End viewing            File =   FULDA3.GPH   Corps Deployments
13:23:27  Start viewing          File =   FULDA3.TXT   Obstacle Summary
13:23:38  End viewing            File =   FULDA3.TXT   Obstacle Summary
13:23:58  Start viewing          File =   FULDA4.TXT   OPFOR Strength
13:24:10  End viewing            File =   FULDA4.TXT   OPFOR Strength
```

file contains text material, the file name ends with the ".TXT" extension; and if the file contains graphics material, the file name ends with the ".GPH" extension. The name of the file contents is the same name that the user sees on a menu when he selects the material for viewing. The menu name is determined by a command in the TGD (Text and Graphics Directory) file, a file which controls the set of text and graphics aids which the user will see. Each text document or graphics screen is contained in a separate file. The TGD directory lists all available aid files for the currently aided problem and associates each file with the name that the user sees on the menu. The TGD file and the overall scheme for presenting aid to the user are described in a separate part.

5.3 LIMITATIONS OF THE JOURNAL

The logic of the journal is general and can be applied to collect other kinds of user statistics as well, but it is currently configured to record the beginning and ending times at which the user views each of the text and graphics aids for the currently aided problem. The journal is not produced if the user works on an unaided problem, i.e., if he elects not to use the problem currently supported by text and graphics displays. The program works normally for an unaided problem, except that no special text and graphics aids are available to the user in analyzing the problem, so no statistics can be collected about the use of these aids.

5.4 UTILIZING THE JOURNAL

The program writes the journal to a file named USERSTAT.DAT. If no such file exists, the program creates it. If a file by that name already exists, the program deletes its contents and starts the journal over for the session. Thus, it is a good idea always to print the journal or save it on diskette immediately after a program session. (If the session was of any experimental interest, that is. The program's action of automatically deleting the previous file's contents allows people to practice with the program without proliferating unneeded journal files. Also, the program does not intrude by asking the user if he wants a journal produced.) The journal is also lost if the program terminates prematurely, say, by the user's issuing an interrupt from the keyboard to escape from the program.

Experimenters can use the journal of a session in several ways, such as those listed below.

(a) Input to Statistical Programs. Because the format of the journal is uniform, and entries are arranged in columns, the journal is apropriate for input to analysis by com puter program. Copies of the journal should be saved on diskette or the hard disk of the XT computer if this kind of use is planned.

(b) Printing the Journal. "PRINT USERSTAT.DAT" is the appropriate DOS command to produce a printed copy of the journal.

(c) Saving on Diskette. Insert a diskette into the floppy drive and issue the DOS command "COPY USERSTAT.DAT A:" to save the journal on the diskette. (The diskette must first be formatted using the "FORMAT A": command. How ever, it is good practice always to be very careful in use of the FORMAT command, as an inadvertent mistake can result in re-formatting the hard disk and losing its contents.)

Other uses include viewing the journal on the monitor and storing the journal in another file on the hard disk of the XT. The latter may be convenient if the copies are named by a systematic convention to ease future reference.

PART III

A GUIDE TO AUGMENTED INFORMATION DISPLAY ON THE PROTOTYPE ANALYST WORKSTATION

1.0 INTRODUCTION

An Augmented Information Display (AID) has been incorporated into PAWS to present textual and graphic information to the user through the color graphics monitor of the workstation. The principal application of the AID is to supplement maps, overlays, and paper-copy texts in presenting tactical vignettes to the user for solution at the work station. PAWS system managers and programmers can design and prepare different sets of texts and graphics for the AID for different scenarios, different hierarchies of decision factors, and different formats for presentation of the material.

This part contains guides for two different kinds of association with the PAWS program. Section 2.0 is a guide for program users, and section 3.0 is a guide for persons concerned with management of programs and files on the IBM PC XT computer.

2.0 THE USER'S GUIDE

This section is addressed to the users of the PAWS decision aid who want to understand how the Augmented Information Display will affect their use of the program.

2.1 BACKGROUND

Past aids, such as ENCOA and DMP, assisted the user in assessing enemy courses of action but did not present information about the battlefield stituation. The burden was entirely on the user to understand the tactical situation through the study of maps, overlays, and textual information, either prior to or during use of the decision aid. That is not the case with PAWS, where one of its new features is the display of information about the battlefield situation during the solution of the problem. While working on a problem, you can view the situation directly on the workstation's monitor.

While you are preparing to respond to the program's requests for information, say, to score an enemy course of action, you can choose to view a set of materials addressing exactly the aspect of the battlefield situation that is of concern. For example, if you are scoring avenues of approach with respect to the disposition of enemy forces, you would get to view information tailored to that factor -- i.e., a description of the enemy force dispositions. What you see on the workstation's screen is not a complete replacement for maps and other printed documents, but it does provide an alternative to depending entirely on your memory, or to interrupting your work to re-read a map or written estimate.

2.2 OPERATION OF THE AID

Operation of the AID is automatic. Once you have learned how to run PAWS, there is no extra work involved in viewing the information displays. At the appropriate points PAWS will either ask you if you want to see a display or will print "H for Help" on the screen, depending on the occasion. In the latter case you have the choice of either continuing with your assessments or entering "H" to see the display before going back to the job at hand. The AID is not intrusive and does not force you to see the information, but it does remind you of its availability.

You will see AID displays in three places in the program: (1) when you begin a problem; (2) each time you score enemy courses of action on another education factor; and (3) when you assess the impact of new intelligence. In the first case you are offered the choice of viewing a general introduction to the tactical vignette under study; in the second case you have the choice of viewing information tailored to the specific evaluation factor; and in the third case you are automatically shown the next message in the stream.

2.3 ACCESS TO THE AIDED PROBLEM

Instructors and experimenters will set up the PAWS workstation with information displays corresponding to different problems at different times, depending on the exercise or experiment in which you are participating. However, there is no more than one aided problem available at one time.

If you need to work on a different problem, you can decline the program's offer to provide the information displays. The access to the AID is discussed below.

2.3.1 THE CREATE MODEL OPTION

When you choose the CREATE MODEL option from the PAWS main menu, you are shown a menu of problem types, including the currently aided problem. PAWS will always assign the aided problem to the first item on the menu. If you elect to work on the aided problem, PAWS will present only the current aided problem. If not, PAWS will disregard the aided problem and will prompt you to describe the problem. In the latter case you are prompted to give the number of enemy courses of action, to name them, and to select a set of evaluation factors -- such as the 25 ENCOA factors -- to be used on your problem. In the case of the aided problem you do not have to make these decisions, since the number of courses of action, their names, and the set of evaluation factors are all set up for you. You are told of these facts after you choose the aided problem.

2.3.2 THE LOAD MODEL OPTION

When you choose the LOAD MODEL option from the PAWS main menu, you do not have make any choice, whether to use the aided problem or not. If the loaded model was originally created using the current aided problem, PAWS will tell you so and will offer to show you an introduction to the problem (along with all other text and graphics for the problem). If the loaded model was not originally created using the current aided problem, no display aid will be offered, and you can proceed to use the model unaided.

2.3.3 ASSESSING THE IMPACT OF NEW INTELLIGENCE

After you have saved a model or loaded a previously created model, you can choose the option ASSESS IMPACT OF NEW INTELLIGENCE from the main menu. After showing you the current probabilities of the enemy courses of action, the program will ask you to name the observation constituting the new intelligence and -- if aid is available -- it also tells you that you can instead enter "H for Help." If you ask for help, you can view a set of informational materials, which may include (depending on the information set up for you to use) a copy of the message describing the new intelligence that you are to assess. After viewing this information, you can name the observation and assess its impact on the probabilities of the enemy courses of action.

2.4 HOW TO VIEW THE SCORING INFORMATION

When you enter "H" (for Help) while scoring an enemy course of action, you will be presented with a menu of documents related to the current evaluation factor. An example is:

```
                  Factor:  Enemy Disposition

          1.  List of Enemy Dispositions
          2.  Map of Enemy Dispositions
          3.  General Discussion of Scenarios
          4.  Schematic of Enemy Avenues
          0.  None of these:  Continue Solving Problem

               Enter choice (0-4): ___________
```

Some of the items offered may be very specific to the factor at hand, while others may be general. To choose the item that you want to view, enter its number and it will appear. If it is a multiscreen document, you will be told "Press RETURN to continue" at the end of each screen, and at the end of the document you will be returned to the menu. If the item is graphic, pressing any key will return you to the menu.

After you have viewed all the items of interest to you, choosing item zero will return you to the scoring process. A similar menu is shown if you ask for help while assessing the impact of new intelligence.

3.0 THE SYSTEM MANAGER'S GUIDE

The purpose of this section is to describe the AID for programmers, experimenters, and others who may need to set up systems of display files for the PAWS program. This chapter assumes that the reader is familiar with basic concepts of the PC-DOS operating system for the IBM PC XT computer.

This section contains an overview of the display aid (subsection 3.1); the structure of the TGD file, which is used to organize the display files (subsection 3.2); how to switch files to different problem (subsection 3.3); and how to prepare new display files (subsection 3.4).

3.1 OVERVIEW OF THE DISPLAY AID

This section describes what the user sees with the display aid (section 3.1.1); distinguishes models, problems, and tree types (section 3.1.2); and summarizes the operation of the augmented information display (section 3.1.3).

3.1.1 WHAT THE USER SEES

In the augmented information display (AID) the user begins his analysis in the usual way. When he sees the tree type menu he is offered a new problem types "O", the default aided problem type. To run an un-aided problem he chooses an old scenario-less problem type, like "PLATOON TREE," in the usual way and perform the usual analysis, in which he specifies the COAs associated with the platoon tree structure, is prompted for scores for each COA at the terminal nodes, and then is asked to assign weights to the nonterminal nodes. To analyze the aided problem he need only select it from the menu and the program will fill in and display (from the current TGD file, whose name will always be "PROBLEM.TGD") the problem's tree type, and its AAs.

After selection of an aided problem, the user sees the introductory material on the scenario and description of the predefined AAs. Then he is prompted for weights and scores, as before, with this important exception: Upon scoring a factor which has been supplied with AID information the user sees the line "type H for help" at the upper right-hand corner of his screen. If he asks for aid he will be shown an AID menu. After he has selected and viewed items from the menu, he can return to providing scores. After he has supplied these he will go on to the next node of the problem tree.

After the user has finished scoring and weighting the nodes of the tree he will be offered the opportunity to store his just-completed model on the disk. If he elects to do this, he makes a record of not only the weights and scores, but the problem type (tree type and scenario) as well. Later, if the user edits the model, the program will check to see if the scenario in effect when the model was created is the same as the scenario currently resident in the subdirectory. If it is, the user will be offered AID for each aided node that he elects to edit; if the old scenario does not match the current one, then the user can still edit the model, but he will not be offered AID.

The user is also offered aid when he assesses the impact of new intelligence on a previously created model. Immediately before updating the COA probabilities he can view a predetermined set of information to help in his estimation of the probability of the observation occurring under the different courses of action. Specifically, when the program asks the user to name the new observation, he can enter "H" (for Help) to gain access to a menu of available information. (Clearly, the user could never give the name "H" to an observation, as that response is reserved to gain entry to the help facility.) Although the user is not forced by the program to view the aid, this location is appropriate for presenting a predetermined message, i.e., new intelligence, to the user for his assessment. In fact, the aid is structured so that a different set of materials can be presented on every message. (That is, there is one packet of materials available for viewing for message number one, another for message number two, and so on.)

3.1.2 NOMENCLATURE FOR PROBLEM TYPES

At this point it will be helpful to distinguish between PAWS models, PAWS problem types, and PAWS tree types. The table below gives the components present in these three different constructs, in order of their decreasing complexity.

	COMPONENTS		
	tree number?	scenario?	weight & scores?
model	YES	YES	YES
problem type	YES	YES	NO
tree type→[1]	YES	NO→[1]	NO→[1]

A model has all three components; it is created and named by the user. An example is 'FULDA 9/3/84.' A problem type is defined by the system manager using a tree utility; examples are 'PLATOON TREE" and 'FULDA.'

The former names a tree type with no associated scenario; the later names a tree type with the "linked" Fulda scenario. Tree types are also created by the system manager, using the tree utility TREEDEF.

3.1.3 THE AUGMENTED INFORMATION DISPLAY UNIT

PAWS has a single, default aided problem that can be selected by the user when he sees the tree type menu; this is problem type 0 on the menu. When the user selects the aided problem, the program consult- the scenario text and graphics directory (TGD) file stored in the current directory; this file contains, among other things, a section that contains the list of AID units available to the user for each node of the tree associated with the aided problem. These units serve to explicate each terminal node of the tree associated with the aided problem.

According to this idea, then, there are two kinds of problem types. The first problem type corresponds to a user-selected tree type; that is, it consists merely of a tree type with no associated scenario. The second -- aided -- kind of problem type consists of the tree type that has scenario information linked to it by the text and graphics directory file. The name of this file is always "PROBLEM.TGD". If the user selects the AIDed problem and there is no TGD file present, the program will terminate, after an error message. Another directory file, "MESSAGES.TGD," can optionally be present to control the presentation of messages during the assessment of new intelligence.

3.2 THE STRUCTURE OF THE PROBLEM.TGD FILE

The TGD file is a text file of parameter lines each of the form

<parameter> = <value> ,

where

<parameter> = SEN | TRE | NAA | LA <i> | SA <i> | NOD | FIL,
<i> in [1, maximum number of COAs (AAs)],
and the symbol "|" is to be read "or".

The value of the SEN parameter is the name of the scenario; the TRE parameter is the name of the associated tree; NAA is the number of avenues of approach; LAi and SAi the name and abbreviation for the ith AA; NOD the node number of the associated tree. The node number of any tree is that determined by its ordinal number among the terminal nodes in the tree display utility.

The FIL parameter is used to specify the name of each file in the sheaf of text/graphics files available to aid the specified node of the tree. A text file is identified by its .TXT suffix; a graphics file by its .GPH suffix. The beginning of a sheaf of these files is delimited by a parameter line containg the number of the tree node, NOD=number. A filename line FIL=filename contains the name of a (text or graphics) file that contains an AID unit; the collection of filename lines between the tree node line and the next constitutes the sheaf of text and graphics files that provide aid for that node. No particular ordering of the parameter lines is required, and blank lines are ignored. Titles for the AID files are written in braces on the right-hand side of the FIL=line; these titles will be used by the program when the menu for a node is created and displayed for each node sheaf. If desired, a title for the node itself can likewise be specified on the NOD=line between braces.

There is a natural order for the lines of the TGD file: the first line is the name of the scenario, e.g., SEN=Fulda; the second its tree name, e.g., (TRE=ENCOA); the AA information and the individual node sheaves in order. An example of TGD file is given below.

```
SEN=FULDA
TRE=ENCOA
NAA=3
LA1=NORTH--MARKSUHL-ALSFELT
SA=NORTH
LA2=SOUTH--DERMBACH-GEDERN
SA2-SOUTH
LA3=SW--LAUTERBACH-ALSFELD
SA3=SO-WE

NOD=0              Introduction
FIL=FULDA0.TXT

NOD=3              Fields of Fire
FIL=FULDA1.TXT     Description of Fields of Fire
FIL=FULDA2.GPH     Map of Fields of Fire

NOD=10             Disposition of US Forces
FIL=FULDA3.GPH     Map--US Forces around Fulda
```

The first and second lines of this TGD file contain the information that the problem defined by this file is composed of the Fulda Scenario paired with the ENCOA tree type. The third line specifies the number of avenues of approach in the scenario (there are three). The next three pairs of lines give the names and abbreviations for the avenues of approach. The next line indicates that the introduction to the problem is contained in the text file FULDA0.TXT. The next line indicates that the parameter lines that follow provide the name(s) of the aid files associated with the node name "Fields of Fire." The names of the files that contain these units are given in the next two lines. These are "FULDA1.TXT," which can be seen from its suffix, is a text unit. The next unit associated with the fields of fire is "FULDA2.GPH:" it is a graphics unit.

The titles indicated for the tree lines in this sheaf indicate that when the AID menu is offered for this node it will be titled "Fields of Fire." This menu will have two items corresponding to the two Aid files, namely "Description of Fields of Fire," and "Map of Fields of Fire."

There are no more units for this node since the next parameter line starts a new sheaf, this time for the "Disposition of US Forces" node. There is only one active graphics unit available, and one can immediately see the names of the tree unit files. It is easy to see that this form is quite flexible and lends itself to rearrangement, insertion, and deletion of AID units.

The connection between the nodes of the scenario tree and the text/graphic sheaves is established by the numbers assigned to the nodes of each tree by the display tree structure facility. In the case of the standard ENCOA tree, for instance, the display looks like

```
Tree structure name - ENCOA
     MISSION ACCOMPLISHMENT
     .  TERRAIN FACTORS
 1)  .  .  Fields of fire
 2)  .  .  Cover and concealment
 3)  .  .  Mobility
 4)  .  .  Seize/deny key terrain
 5)  .  .  Observation
 6)  .  .  Natural/artificial obstacles
     .  U.S. FORCE FACTORS
 7)  .  .  Disposition
 8)  .  .  Strength and condition
 9)  .  .  Reserves
10)  .  .  Logistics support
11)  .  .  Probable actions/reactions
12)  .  .  Command and control
```

. OPFOR FACTORS
13) . . Current disposition
14) . . Strength and condition
15) . . Reserves
16) . . Logistics Support
17) . . Command and control
. WEATHER FACTORS
18) . . Observation/visibility
19) . . Cover and concealment
20) . . Mobility
21) . . Extreme weather effects
. RISK FACTORS
22) . . US actions/reactions
23) . . Dependence on other commands
24) . . Dependence on surprise/deception
25) . . Unexpected weather

3.3 THE STRUCTURE OF THE MESSAGES.TGD FILE

A directory file, MESSAGES.TGD, associates with each message a sheaf of files containing the message text together with whatever aid files go along with it, if any. The message TGD file is optional, and it is permissible for there to be a problem TGD file, offering aid for the creation of a model, without there being any message TGD file present to aid in the assessment of new intelligence.

The structure of the message TGD file closely follows that of the PROBLEM.TGD file, with the following exception: File packets for each message are delimited by the parameter line

MSG = <message number>

and the next MSG line. The scenario must be identified with a

SEN = <scenario name>

line, so that the display aid logic can determine whether or not the problem TGD file and the message TGD file refer to the same scenario.

An example of what the user sees upon requesting help when assessing the impact of new intelligence is:

```
Menu for Message  n

(1)  Message Text
(2)  Introduction to Scenario
(3)  Relevant Text File
(4)  Relevant Graphics File
(0)  Return to main menu
Enter choice (0-4)
```

The user sees such a menu if he asks for help instead of naming the current message, the first step he must accomplish to assess the impact of new intelligence. The program places the message "H for Help" in the upper right-hand corner of the screen to remind the user of availability of text and graphics aid for the current message.

The systems manager can provide for user access to any of the PAWS text and graphics files in the current directory. He need merely include them in a message sheaf in MESSAGES.TGD. He should also check for identity of scenario name between the message and problem TGD files.

It is not required that a message be present in the sheaf of files associated with a message. Such a situation would be convenient when the user has a paper copy of the message and is using the computerized aid only to review a text or graphics file that might help him interpret the message. In fact, the message TGD file can even be incomplete, i.e., can skip one or more messages in the message stream, and offer no aid at all for them. (There would be no MSG line in the file for that message.) An example extract from a message TGD file follows:

```
SEN = JAYHAWK

MSG = 01            MESSAGE 1 OF 18
FIL = MSG1.TXT      MESSAGE TEXT
FIL = JAYO.TXT      SCENARIO SUMMARY

MSG = 02            MESSAGE 2 OF 18
FIL = MSG2.TXT      MESSAGE TEXT
FIL = JAYO.TXT      SCENARIO SUMMARY

MSG = 03            MESSAGE 3 OF 18
FIL = MSG3.TXT      MESSAGE TEXT
FIL = JAYO.TXT      SCENARIO SUMMARY
```

As with the problem TGD file, the comments inside braces are shown to the user on the aid menu.

3.4 HOW TO CHANGE AIDED PROBLEMS

Because of the number and size of files needed to run the PAWS program, it is convenient to set aside an entire subdirectory of the XT hard disk for the PAWS program and data files. A convenient way to change the files from one problem to another is to erase all files having TGD, TXT, and GPH extensions to their names (DEL *.TXT, for example) from the subdirectory and then to copy the new TGD, TXT, and GPH files into the subdirecory. Each problem can be stored on one or more floppy diskettes and copied onto the hard disk when a change of problems is desired. The PAWS program is then run from the hard disk. (A hard disk is necessary because two floppy diskettes cannot hold all the program and data files.)

3.5 HOW TO PRODUCE NEW DISPLAY FILES

Three kinds of display files are involved in describing an aided problem: those having names ending in the extensions TGD, TXT, and GPH The sets of files, describing new problems, can be prepared as follows:

(1) The TGD Files. The two TGD files are text files and can be prepared by use of a text editor, such as the IBM PC-DOS EDLIN program or a word processor program.

(2) The TXT files. These are also text files and can be prepared with a text editor. When editing these files, it is useful to remember that the display routines present 22 lines of text at a time.

(3) The GPH files. The graphics files are, in the terminology of the PC-DOS operationg system, direct access binary files. Each GPH file describes a single frame shown on the color graphics monitor of the IBM PC XT. These files are built in a special format used by a separate picture-composition program, named ARTIST. New picture files produced by ARTIST can be assembled and used as the GPH files for a new aided problem.

PART IV

TREEDEF: TREE UTILITY FOR DEFINING AND EDITING PROBLEM STRUCTURES

TREEDEF: TREE UTILITY FOR DEFINING AND EDITING PROBLEM STRUCTURES

To reduce the size of the PAWS (formerly ENCOA-BAUDI) program, the Tree Utilities have been made a separate utility program. The utilities are used to manipulate the hierarchies of factors and categories (i.e., the tree structures), which are then used in the PAWS program to assess the enemy course of action.

The file TREEFILE.DAT is used to store the tree structures. The file can contain the descriptions of up to nine different problem structures. (Each description specifies the names of the factors and categories and their structure.) The PAWS program references this file to give the user a choice of structures to act as a template in the creation of the user's model. TREEDEF can be used to change TREEFILE.DAT, i.e., to create new trees, delete old ones, and edit the contents of existing trees. The casual user of PAWS need not know of this file, and only experimenters and system managers need know how to use the TREEDEF utility.

It is recommended that TREEDEF reside in the same subdirectory of the XT's hard disk as the PAWS program. In that way it will always be acting on the same version of the tree file currently being used by PAWS. TREEDEF must also consult the MODELS file before deleting a tree structure in order to assure that no existing model references the tree. A model cannot be interpreted without reference to its corresponding tree structure, and this check is necessary to maintain consistent data structures (i.e.,to avoid fatal errors).

TREEDEF offers the same options as the former Tree Utility menu of PAWS:

(1) Create Tree Structure

(2) Display Current Tree's Outline

(3) Edit Current Tree's Structure

(4) Save Current Tree

(5) Display Tree Types

(6) Get (Load) Tree Types

(7) Delete Tree Type

(8) Initialize Tree File

(0) Exit Program

(1) <u>Create Tree Structure</u>

Leads user through the creation of new tree structures. Begins by asking user to list names of top-level categories, and then prompts for names of subordinates of each of them. An abbreviated tree structure is displayed in a window at the top of the screen, so that the user can always see how much he has done and where he is adding into the tree.

(2) <u>Display Current Tree's Outline</u>

If a tree has been created or loaded, this option shows its structure in outline form, by indenting different levels of the tree.

(3) <u>Edit Current Tree's Structure</u>

Allows user to (a) change names of tree nodes (factor and category names); (b) delete nodes from the tree; (c) add new nodes to the tree; and (d) examine the results of these changes.

(4) Save Current Tree

Puts a newly created or edited tree structure (i.e., the "current tree" in TREEDEF) into the tree file, where it can be stored for future use by TREEDEF and the PAWS program.

(5) Display Tree Types

Displays a list of the names of the trees available in the TREEFILE.DAT file.

(6) Get (Load) Tree Type

Retrieves a tree from the tree file, a necessary first step before displaying the tree or editing it.

(7) Delete Tree Type

Erases a tree permanently from the tree file. Not permitted if a model using this tree currently exists.

(8) Initialize Tree File

Destroys all old tree information in the file and makes all records available fur storing new tree structures.

(0) Exit Program

Stops execution of the tree utility.

PART V

A USER'S MANUAL FOR THE ARTIST PROGRAM

1.0 INTRODUCTION

The purpose of this part is to act as a guide to the use of ARTIST, a graphics composition program for the color graphics monitor of the IBM PC XT computer. No special computing expertise is needed to understand this guide, but readers may find it useful to be familiar with the keyboard of the XT computer.

ARTIST was developed to serve as a means of developing graphics display for use in the Prototype Analyst Workstation (PAWS). PAWS utilizes elements of multiple-attribute utility theory and Bayesian probability theory in a set of procedures for assessing the probabilities of alternative enemy courses of action, for example, as a tool for intelligence analysts in battlefield situation analysis. As one of its options, PAWS allows the user to view a set of text and graphics displays explaining the combat scenario of a predetermined problem on the color graphics monitor of the XT computer. At a given time, only one such "aided" problem is available to PAWS. However, by changing the text and graphics files residing with the PAWS program on the hard disk of the XT computer, an experimenter or system manager can change the scenario that users of PAWS are to analyze. Through the use of a text editor (to prepare the text files) and through the use of the ARTIST program (to prepare the graphics files), an experimenter can prepare new scenarios and save them (on diskette, for example) for future use with PAWS. The text and graphics system for PAWS, including the organization of separate text and graphics files through a directory file, is explained in part III above.

2.0 BACKGROUND

The ARTIST program is designed to support the graphics needs of the PAWS program on the IBM PC XT computer and embodies many design features dictated by its interface to PAWS and by the characteristics of the computer's color graphics monitor. Section 2.1 describes the interface to PAWS, and section 2.2 describes the relevant features of the color graphics monitor.

2.1 INTERFACE TO PAWS

The following discussion is for readers familiar with the DOS operating system of the IBM PC computer. Other readers may wish to skip this section of the guide. However, any user interested in setting up a set of files to describe a new scenario to PAWS may also find it useful to understand the interface between ARTIST and PAWS.

PAWS looks to two files, PROBLEM.TGD and MESSAGES. TGD, to find the names of the files containing the text and graphics supporting its current scenario. The three-letter extension on the end of the file name (i.e., TGD) was chosen by the program developers to stand for "text and graphics directory." These two files are just that: directories to the text and graphics files associated with the scenarios. More specifically, PROBLEM.TGD associates files with the factors of an evaluation hierarchy (such as ENCOA), and MESSAGES.TGD associates files with intelligence updates. That is, there is a packet of explanatory text and graphics files associated with each evaluation factor and each updating message. (Each file can be referenced more than once.)

In the case of graphics, one file contains one picture, i.e., one screen-full of information. PAWS distinguishes between text and graphics files by the three-letter extension at the end of the file name, and separated from the file name by a period. The extension for a text file is "TXT", and the extension for a graphics file is "GPH". The name itself can consist of from one to eight alphanumeric characters; i.e, they can be any legitimate DOS file name. For example, AREAMAP.GPH is graphics file, and WEATHER.TXT is a text file.

ARTIST can store up to 15 pictures at a time in files named PICFILxx.DAT, where xx ranges from 01 to 15. After one or more pictures have been completed, the ARTIST user can copy its corresponding DAT file into another file for use by PAWS. If the ARTIST program's menu shows that a picture named "Disposition of Enemy Forces" is in the 12th file, then the user might copy it into a file named REDUNITS.GPH for permanent storage. That is, after exiting ARTIST, he might insert a formatted diskette into the disk drive and issue the DOS command

```
COPY  PICFIL.12.DAT  A: REDUNITS.DAT
```

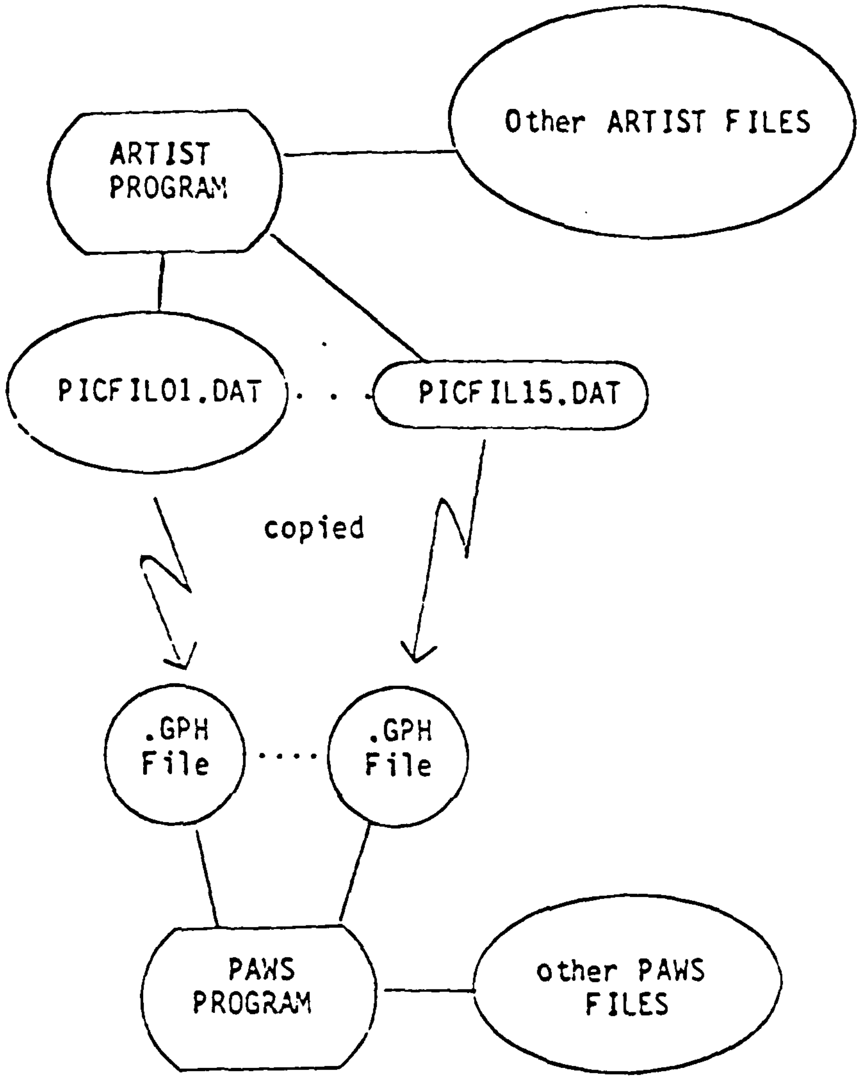
He might then label the diskette with the name of the scenario and put it aside to accumulate and save a set of text and graphics for that scenario.

Exhibit 2-1 illustrates this relationship between PAWS and ARTIST.

2.2 THE COLOR GRAPHICS MONITOR

ARTIST is designed to work with the IBM color graphics monitor, or other monitors that work with the color graphics adapter of the IBM PC computer. These devices fix the resolution of the picture and the colors available on one picture.

EXHIBIT 2:1 INTERFACE BETWEEN ARTIST AND PAWS



ARTIST operates in color graphics mode, which has a resolution of 320 pixels horizontally by 200 pixels vertically.[1] The graphics area of the screen is a square; so that eight pixels horizontally have the same length as five pixels vertically.

Each pixel can have one of four colors. The choice of color depends on which of two palettes is chosen, as shown in the table below:

	PALETTE 0	PALLETE 1
Color 0	Background	Background
Color 1	Green	Cyan
Color 2	Red	Magenta
Color 3	Yellow	White

The background color can take on any one of 16 values, as listed below:

0	Black
1	Blue
2	Green
3	Cyan
4	Red
5	Magenta
6	Brown
7	White
8	Gray
9	Light Blue
10	Light Green
11	Light Cyan
12	Light Red
13	Light Magenta
14	Yellow
15	High-intensit. White

The palette defaults to palette zero (green-red-yellow), and the background color defaults to dark grey in ARTIST. The user can alter these values while running ARTIST, but the palette and background color are fixed in PAWS: A black background is used in PAWS because it gives the sharpest contrast, and the colors in palette zero have proven to be more easily distinguished than those in palette one.

[1] In black and white graphics mode the resolution is 640 pixels horizontally by 200 pixels vertically.

3.0 THE MAIN MENU

The overall operation of ARTIST is controlled through a menu, which controls such top-level functions as creating new pictures and retrieving previously-created pictures from a file. When ARTIST begins, it displays the following menu:

```
                    MENU OF ARTIST OPERATIONS

          1) Define feature
          2) Clear picture
          3) Retrieve picture from file
          4) Define symbol
          5) Delete symbol from symbol file
          6) Initialize symbol file (debug)
          7) Dump symbol file
          0) Exit current menu

         Enter feature operation number (0 - 7) :
```

The usual sequence of operations in creating a new picture is as follows:

(a) Choose item one from the main menu and follow the procedure for creating a new picture. (This procedure is described in detail below.)

(b) After the picture is finished, the program will ask "do you want to save the picture?" If you do, answer "Y" and the program will store the picture for future reference.

(c) You are then shown the main menu again. At this time, you can select item two, "Clear Picture," if you want to draw a new picture.

(d) If you want to draw a new picture, begin the process over with step a, above. Otherwise, choose item zero to exit the ARTIST program.

Thus, the basic sequence of operations is: clear the screen, define a picture, and store it.

The ARTIST program "remembers" the last picture that you created until you either clear the screen, retreive another picture from a file (item 3 on the main menu), or exit the program. By remembering a "current picture," it makes easier the process of creating a new picture, storing it, and then adding more detail to the picture before storing a new version of it. This process can be useful when creating several related pictures, e.g., in showing an unfolding sequence of information.

After creating and saving a picture, there are three things that could be done to the contents of the "current picture" before beginning to create a new picture:

(1) Leave the "current picture" alone, because it is the starting point for the next picture;

(2) "Clear picture" (item two) to start the next picture from a blank screen; or

(3) "Retrieve picture from file" (item three), if an existing, stored picture is to be the point of departure for the next picture.

When going back into item one to define a new picture, the program retains the contents of the "current picture," which can be added to (or written on top of) as the new picture is drawn. This process is analogous to that of using a clear canvas or a used one when painting a picture.

The main menu also gives access to a facility for defining and storing user-designed symbols. These symbols can be recalled later and reproduced as part of a picture composition. Symbol definition is defined in a later section.

4.0 DEFINING A PICTURE

Once item one has been selected from the main menu, the program displays a graphics screen, which will either be blank or display a previously created picture (depending on the contents of the "current picture"). A cross-hairs cursor will also be shown in the middle of the screen. Various drawing operations are defined with respect to the pixel at the center of the cursor, which can be moved around the screen.

The process of defining a picture involves moving the cursor around the screen, coloring individual pixels, typing characters on the screen, filling regions, and placing predefined symbols on the screen. These functions are described in later sections of this guide.

Creating a picture begins in "draw mode," which is the basic mode of sketching a new picture. Various specialized modes of operation are accessible from draw mode, and after exiting these specialized modes the program returns to draw mode. These subordinate modes are:

(a) Alphanumeric mode, for typing of text;

(b) Symbol mode, for predefined symbols reproducing on the screen;

(c) Alter video values mode, for altering the palette, the background color, and the color of the cursor;

(d) Fill mode, for painting the inside of closed figures; and

(e) Backup mode, which lets the user move the cursor backward and forward through the feature he has drawn on the screen, for re-coloring or erasing mistakes.

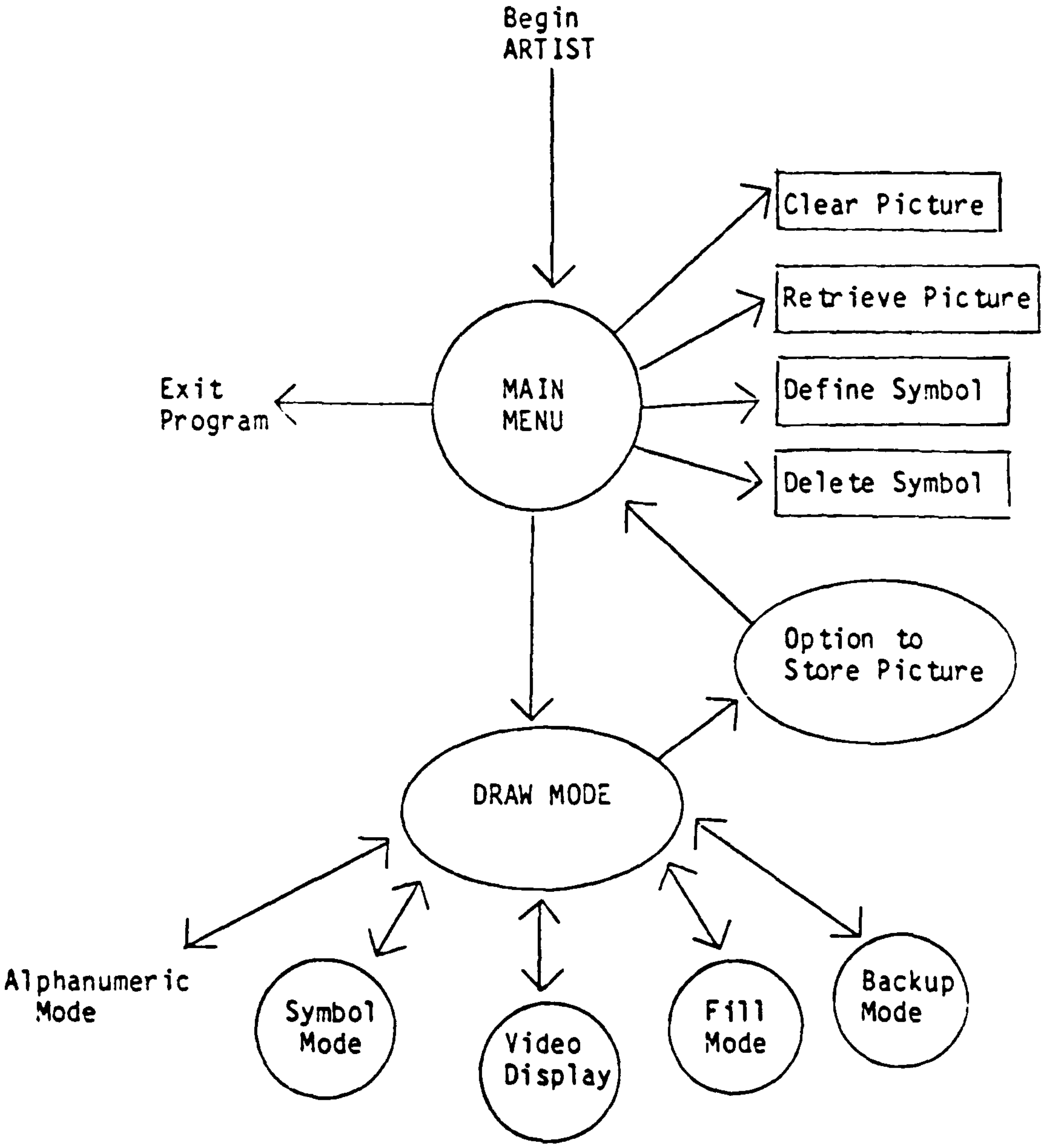
The process of creating a picture consists of composing a freehand sketch in draw mode, augmented by the specialized functions of the other modes, repeated as many times as needed. The overall process of creating a picture is shown in exhibit 4-1, along with the other, main menu functions of the ARTIST program.

Although they perform different functions, the ARTIST modes all share common procedures for their operation. Because each mode offers a number of alternative commands, and since the same key may mean different commands under different modes,[1] a help facility exists to remind the user of the commands available in each mode. Pressing the RETURN key toggles the screen back and forth between graphics mode (in which the user's drawing is displayed) and text mode (in which a help screen is displayed). The help screen gives the name of the current mode, lists the valid keystrokes in that mode, and defines the operation corresponding to each valid keystroke. The user can toggle to the help screen at any time and toggle back to the graphics screen to execute the command or commands desired. ARTIST acts as a sketchpad, in that it executes the user's commands and displays their cummulative results on the screen as they are entered.

In general, draw mode and the specialized graphics modes respond to single keystrokes. That is, the command is executed as soon as the user presses the corresponding key, and it is not necessary to press the RETURN key. In fact, the RETURN key has the special function of toggling between the graphics screen and a help screen, as already described. The ESC key and the Q key have the same function in most modes: "escaping" the mode, or "quitting" it to go to the mode from which it was called. The only exception is Alphanumeric mode, on which the RETURN key has this function. (The Q key has to be reserved to type the letter "Q", and the

[1] This multiplicity is a necessity, because there are more commands then there are keys.

EXHIBIT 4-1: SCHEMATIC OF ARTIST OPERATION



RETURN key is a familiar way of signaling the end of a line of text on a computer keyboard.)

On a few special occasions, when an erroneous single keystroke might be especially catastrophic, the user must press the RETURN key before the program will react to the user's instruction. These special occasions (on the Main Menu and in selecting fill parameters in Fill Mode) require the user to answer a direct question or filling in a blank in a prepared form, and it is evident to the user when these occasions arise.

5.0 DRAW MODE

Selecting item one (define Feature) from the Main Menu causes ARTIST to enter Draw Mode and switches the screen to graphics mode.

The user can then enter commands from the keyboard and see the resulting sketch developing on the screen. Exhibit 5-1 lists the commands recognized in Draw Mode. The exhibit is a reproduction of the help screen available in Draw Mode. There are four categories of Draw Mode Commands: feature definition options, changing pen colors, cursor movements, and "other" commands.

5.1 FEATURE DEFINITION OPTIONS

These command control the selection of the various specialized graphics modes and the up/down position of the imaginary pen located at the center of the cursor. Each command is a single letter, in either upper case or lower case.

A - go to alphanumeric mode;
B - go to backup mode;
D - put the pen down (i.e., into position for drawing);
F - go to fill mode;
S - go to symbol mode;
U - put the pen up (i.e., above the "paper" to disable drawing; a user would typically put the pen up before moving the cursor to a new position without writing on the intervening pixels); and
V - go to video mode (to change default video values).

When entering Draw Mode from the Main Menu, the program puts the pen in the "up" position. To begin drawing on the screen, the user would press "D" before moving the cursor. The pen controls, as well as several other useful commands, are also available in the other modes of ARTIST operation.

EXHIBIT 5-1: HELP SCREEN FOR DRAW MODE

```
Feature definition options:

     A - alphanumeric mode (graphics)    S - symbol mode
     B - backup mode                     U - pen up
     D - pen down                        V - alter video default values
     F - fill mode


Pen colors:

     E - erase (background color)
     C - cyan                 M - magenta      W - white
     G - green                R - red          Y - yellow


Cursor movement:

     7 - ne                   8 - n            9 - nw
     4 - w                    5 -              6 - e
     1 - sw                   2 - s            3 - se


Other:

     Q, <esc> - exit drawing mode
     <CR>     - toggle between draw mode menu and graphic screen
```

5.2 PEN COLORS

These commands change the color with which the pen writes on the screen. When the program enters Draw Mode from the Main Menu, it sets the pen color to the background. (Drawing in background color is not visible against the background, of course, but it can be used to trace features inside colored areas. The pen color commands are:

E - color zero (background color);
G or C - color one (green or cyan);
R or M - color two (red or magenta);
Y or W - color three (yellow or white).

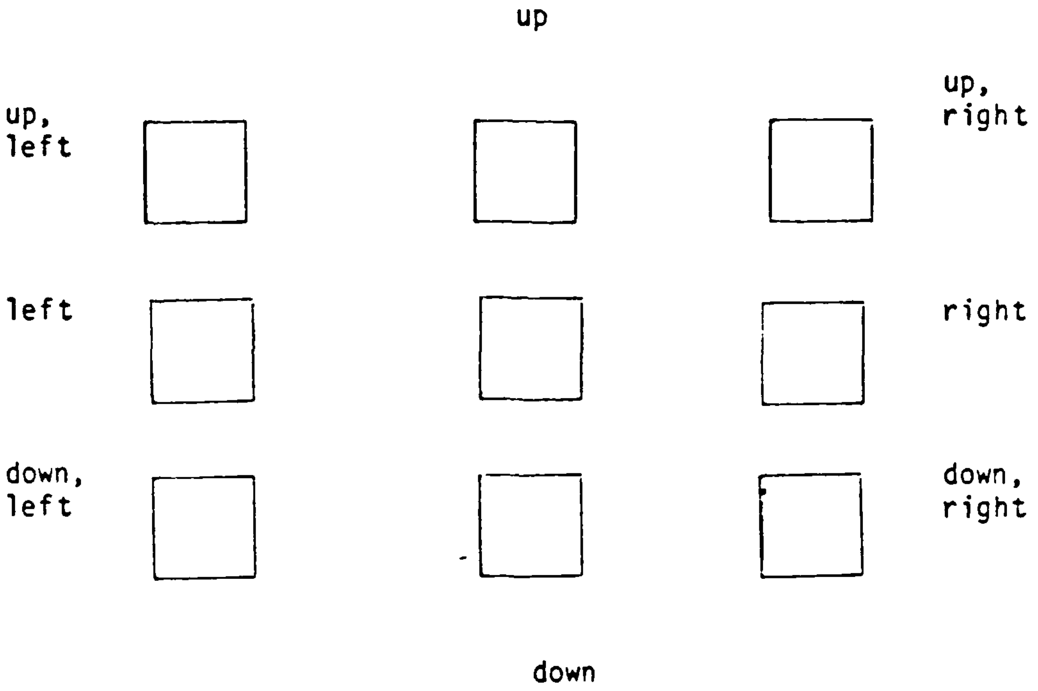
These commands select among the four colors available with the currently chosen palette. With palette zero these are background, green, red, and yellow; with palette one these are background, cyan, magenta, and white. The choice of palette can be changed only in Alter Video Values Mode; it is initially set to palette zero (green-red-yellow). Independently of the palette, either G or C can select color one, either R or M can select color two, and either Y or W can select color three.

5.3 CURSOR MOVEMENT

The numerical keys 1-4 and 6-9 control the movement of the cursor. These keys are arranged in the numerical keypad, on the right side of the keyboard, in a convenient pattern for control of cursor movement one pixel in any direction: vertically, horizontally, or diagonally. The keyboard pattern is shown in exhibit 5-2.

Above the numeric keypad there is a key marked NUM LOCK, which contr.ls the interpretation of the numeric keys: pressing the NUM LOCK key toggles the meaning of the keys between (a) the digits one through nine and (b) the arrows and other special symbols that are also written

EXHIBIT 5-2: USE OF THE NUMBERIC KEYPAD FOR CURSOR MOVEMENT



on the keys. If pressing the keys of the numeric keypad does not move the cursor, the NUM LOCK key should be pressed (once) to make the computer interpret the keys as the numerals. Thus, the NUM LOCK key acts somewhat like an upper/lower case lock, but it acts only on the numeric keypad (and on some keys in the keypad's vicinity not of concern here).

Once the keypad is locked on, depressing one of the keys (except number five, which is unused) will cause the cursor to move one pixel in the indicated direction. If the pen is in the down position, the pixel to which the cursor moves is colored to the current pen color. If the pen is in the up position, the cursor moves, but the destination pixel does not change color. The cursor consists of cross-hairs with a "hole" in the middle, to allow viewing of the color of the pixel on which the cursor is centered. Holding down one of the numeric keys causes the cursor to continue moving in the indicated direction at a constant rate (a rate slow enough to allow user control of the distance of movement, but fast enough to facilitate the drawing of straight lines without undue delay).

The top and side of the screen act as boundaries to drawing. That is, the program simply does not respond to attempts to move the cursor beyond the edge of that screen. The cursor does not "wrap around" and move to the opposite edge of the screen.

The numeric keys on the top row of the keyboard can also be used to control cursor movement, but the numeric keypad is arranged in a much more convenient configuration.

5.4 THE "OTHER" KEYS IN DRAW MODE

The carriage return key is denoted on the help screen by the symbol <CR>.

The carriage return key is used to toggle back and forth between the graphics screen and a help screen, listing the valid keystrokes in Draw Mode. Denoted by the <CR> symbol on the help screen (and in this guide), the carriage return key is marked by an arrow with a right-angle turn in its shaft (differently from the keyboards of other computers, where the word RETURN is used as a label).

5.5 SUMMARY OF DRAW MODE

The basic purpose of the Draw Mode is to assist in making freehand sketches and to act as a "home mode" from which the specialized modes are reached. A typical sequence might consist of putting the pen "up", moving it to a certain area of the screen, putting it "down," and drawing one or more pixels of a desired color. The coarse texture of the screen (320 by 200 pixels in color graphics) makes it practical to step through the more detailed parts of a feature pixel by pixel to achieve a desired pattern. More regular parts of a feature can be created with the commands accessible in other modes, in which a single keystroke can generate patterns coloring many pixels.

6.0 SUBORDINATE MODES

This chapter explains the operation of the subordinate modes entered from Draw Mode. These modes, which achieve specialized visual effects on the color graphics monitor, are alphanumeric mode (section 6.1), symbol mode (section 6.2), alter video values mode (section 6.3), fill mode (section 6.4), and backup mode (section 6.5). Each of these modes is subordinate in the sense that terminating the mode's operations returns control to Draw Mode.

6.1 ALPHANUMERIC MODE

Alphanumeric mode is one of the specialized modes accessed from Draw Mode. Its purpose is to allow the typing of character information on the screen at the cursor location. Any of the characters defined on the keyboard (letters, both upper and lower case, numerals, punctuation marks, and other special characters) may be used.

Typing continues in alphanumeric mode until the carriage return is entered. This is the one exception to the general rule that <CR> sig nals a help screen, and ESC and Q signal a return to Draw Mode. The Q is needed as a typing character, and the carriage return is not a printing character. For this reason, as well as the fact that it is a usual way of ending a line of text in other applications, the carriage return signals the end of alphanumeric mode. No help screen is available in alphanumeric mode, but the keys all have fairly natural interpretations in this mode.

ARTIST generates characters the 40 column by 25 line format that the color monitor supports when it is displaying color graphics. When a key is pressed, ARTIST places the corresponding character into the ceil occupied by the cursor (the center of the cursor, specifically). That is, the character appears in that cell of the 40 by 25 matrix into which text is generated. (Thus, each cell occupies eight pixels horizontally by eight pixels vertically).

After a character is typed, ARTIST advances the cursor eight pixels to the right (unless it encounters the edge of the screen) and waits for another character. When it reaches the rightmost column, ARTIST leaves the cursor there (and will accept new characters to replace any already there) until it receives a carriage return to go to Draw Mode. Backspacing is not supported, nor is automatic spacing to the next line. The purpose of ARTIST is to generate freehand graphics, and character generation was viewed (by the program's developers) as only an occasional part of a picture's composition. However, by existing alphanumeric mode and repositioning the cursor in Draw Mode, a user can position text anywhere on the screen (e.g., to correct previous errors in typing characters).

Alphanumeric mode types text in the current color of the cursor. The color of the cursor can be changed in alter video values mode.

6.2 SYMBOL MODE

The purpose of symbol mode is to overlay symbols previously designed and stored with the Define Symbol option from the Main Menu of ARTIST. This feature of ARTIST allows a user to prepare a repertoire of special-purpose symbols, such as unit symbols or stylized cultural features for maps, and then to incorporate them repeatedly into different pictures by using symbol mode. Upon pressing "S" to enter symbol mode, the user sees a menu of available symbols, for example:

```
Symbols available for selection:

     0)  None
     1)  Motor Rifle Division
     2)  Tank Division
     3)  Town

Enter symbol number (0-3): ____
```

After the user enters a number (and presses the carriage return key), ARTIST returns the user's picture to the screen, at which time the user can position the cursor and specify where the symbol is to be drawn.

Exhibit 6-1 displays the commands recognized by ARTIST during symbol mode. These commands allow the user to draw and erase the symbol, move the cursor to positions where the symbol is to be drawn, and change the color of the symbol. As with most other modes, the carriage return switches to and from a help screen, and the Q and ESC keys cause ARTIST to leave symbol mode and return to Draw Mode.

As soon as the user issues the command to draw the symbol, ARTIST draws the symbol at the cursor position. If the symbol is not positioned properly, it is not necessary to erase it explicitly. Simply adjusting the cursor position and issuing the draw command again cause the symbol to be erased and moved to the new position. The symbol does not become a "permanent" component of the picture until ARTIST leaves symbol mode. (For this reason, only one copy of the symbol can be incorporated into the picture on each entry to symbol mode).

When the user presses Q or ESC to leave symbol mode, ARTIST erases the symbol and draws it again at the same location. The process is visible to the user and, while it may seem redundant, it is really a consequence of ARTIST's ability to erase and reposition symbols. Until leaving symbol mode ARTIST draws the symbol in a way that facilitates erasure without disturbing features over which the symbol is drawn and erased.

EXHIBIT 6-1: HELP SCREEN FOR SYMBOL MODE

```
Symbol Mode Operations:

     D - draw symbol at cursor location
     E - erase current symbol
     S - select current symbol from menu


Symbol Colors:

     E - erase (background color)
     C - cyan                M - magenta         W - white
     G - green               R - red             Y - yellow


Cursor Movement:

     7 - ne                  8 - n               9 - nw
     4 - w                   5 -                 6 - e
     1 - sw                  2 - s               3 - se


Other:

     Q, <esc> -  exit drawing mode; save symbol as part of
                 feature
     <CR>     -  toggle between symbol mode menu and graphic
                 screen

Enter symbol mode option:
```

This is the reason that the intersection of a symbol and an existing feature of another color is drawn in a third color -- until symbol mode terminates and ARTIST redraws the symbol entirely in the true color.

6.3 ALTER VIDEO VALUES MODE

The purpose of alter video values mode is to allow the user to choose a combination of background color, palette, and cursor color. Section 2.2 above lists the allowable options for palettes and colors with the color monitor of the PC computer. The choices of palette and background color apply to the current picture only at the time it is being created with the ARTIST program. The values can be changed any number of times while the picture is being created, but displayed by the PAWS program (their eventual destination), a black background and the green-red-yellow palette are always used (to produce the greatest contrast). Exhibit 6-2 shows the commands recognized in this mode.

6.4 FILL MODE

The purpose of fill mode is to paint the interior of figures with solid colors or colored patterns. The interior of any region enclosed by a closed boundary of a single color can be filled, but it is important that the boundary be closed: Any break in the boundary can allow the fill algorithm to "leak out" of the interior and ohliterate the contents of the screen.

Fill mode operates in two steps: selecting the attributes to be used in the fill and identifying the region to fill. Upon entry to fill mode ARTIST prompts the user to select the fill attributes (fill color, boundary color, and fill pattern) from the menus shown in exhibit 6-3.

EXHIBIT 6-2: HELP SCREEN FOR ALTER VIDEO VALUES MODE

```
Menu of Alter Video Display Default Value Options

      B:  Step through Background colors

      C:  Step through Cursor colors

      H:  Help; text menu of alter video display default value mode key
          strokes

      P:  Step through Palette colors

      Q:  Quit; return to Draw Mode Menu

    Esc:  Escape; same as Quit

  Space:  Toggle between color graphics and text (menu) modes

Enter alter video default value option:
```

EXHIBIT 6-3: SELECTION OF FILL ATTRIBUTES

```
Selection of fill attributes


Fill colors:

     (0)  Background color
     (1)  Green               (cyan)
     (2)  Red                 (magenta)
     (3)  Yellow              (white)

Enter fill color number (0 - 3):


Boundary colors:

     (0)  Background color
     (1)  Green               (cyan)
     (2)  Red                 (magenta)
     (3)  Yellow              (white)

Enter boundary color number (0 - 3):


Fill patterns:

     (1)  Solid
     (2)  Horizontal stripes
     (3)  Vertical stripes
     (4)  Right handed diagonal stripes
     (5)  Left handed diagonal stripes
     (6)  Sparse dots

Enter fill pattern number (1 - 6):
```

Upon completion of these choices, ARTIST switches to the user's graphics picture, so that the user can move the cursor and identify the region to be filled. Pressing the "F" key signals ARTIST to execute the fill, which it initiates from the current cursor position. Before leaving fill mode, any number of regions can be filled, and the choice of fill attributes can be altered by pressing the "S" key to select new values. Exhibit 6-4 shows the commands available in fill mode.

6.5 BACKUP MODE

ARTIST mainains a list of the pixels colored by Draw Mode operations, in the order colored, up to a limit of 2000 pixels. (After 2000 pixels, it deletes the oldest pixels from the history to keep a record of the most recent pixels colored.) This history is kept for the backup mode, which moves the cursor backward and forward through the pixel history. The user need only hold down the "B" or "F" key to move the cursor backward or forward through the history, respectively,. Exhibit 6-5 shows the available commands in backup mode.

ARTIST maintains the history only until exiting Draw Mode. The history is not stored with the picture contents when saving the picture in a file for future use. (Recall, when exiting Draw Mode, ARTIST asks the user if the picture is to be saved on disk.) If the picture is retrieved later, and more features added, ARTIST starts its history anew and supports backup mode only over the set of pixels drawn since picture retrieval.

Backup mode is particularly useful for correcting freehand drawing errors occurring in Draw Mode and during symbol creation. By putting the pen in the "down" position and selecting the background color, the user

EXHIBIT 6-4: HELP SCREEN FOR FILL MODE

```
Fill Mode Operations:

     F - fill bounded region
     S - select fill pattern, color, and boundary

Cursor Movement:

     7 - nw                   8 - n            9 - ne
     4 - w                    5 -              6 - e
     1 - sw                   2 - s            3 - se


Other:

     Q, <esc> - exit fill mode; save filled region as part of
                picture
     <CR>     - toggle between fill mode menu and graphic
                screen

Enter fill mode option:
```

EXHIBIT 6-5: HELP SCREEN FOR BACKUP MODE

```
Backup Mode Operations:

     B - backup through feature's pixels
     F - move forward through feature's pixels


Backup Colors:

     E - erase (background color)
     C - cyan                  M - magenta     W - white
     G - green                 R - red         Y - yellow


Other:

     Q, <esc> - exit symbol mode; save symbol as part of
                feature
     <CR>     - toggle between symbol mode menu and graphic
                screen

Enter backup mode option:
```

can erase the pixels most recently drawn by backing up the cursor over them. By putting the pen in the "up" position, he can back up to an earlier portion of the history before putting the pen down and erasing or changing colors.

7.0 OTHER MAIN MENU FUNCTIONS

Sections 3 and 4 have already discussed the main menu, mainly from the perspective of creating a new picture. This section discusses the other options on the main menu: picture retrieval (section 7.1), symbol creation (section 7.2), and symbol deletion (section 7.3).

7.1 PICTURE RETRIEVAL

When ARTIST enters Draw Mode from the Main Menu, the contents of the graphics screen are determined by the following cases:

(a) If the user has selected the Erase Screen option from the Main Menu immediately prior, picture is blank.

(b) If the user has retrieved a picture from those saved previously, that picture will be on the screen. He can draw over it and later store the combined results.

(c) If the user has been in Draw Mode previously, and has not cleared the screen or retrieved another picture, ARTIST displays that picture. It always remembers this picture until the user exits ARTIST or takes some action to alter the"current" picture.

(d) When ARTIST begins operations, it clears the "current" screen, so that the user can start with a blank screen.

When a picture is to be retrieved, ARTIST displays a list of names of 15 stored pictures, of which one can be selected for retrieval as the current picture.[1] Retrieving pictures, and then switching to Draw Mode to view them, is useful for reviewing the results of recent work on ARTIST. For example, locating a picture's position on the list is the first step in copying a file for use with the PAWS program. Also, it is useful when determining which slots are expendable and can be replaced with new pictures.

7.2 SYMBOL DEFINITION

This main menu item is used to draw features and store them for future retrieval and overlaying on pictures. Symbols are stored in a file separate from that used for pictures. The term "picture" refers to a full screen of information, while a "symbol" refers to a limited set of pixels. When redrawn on the screen, the symbol can be placed at selected position (indicated by the cursor, under user control).

Upon entry to symbol definition, ARTIST clears the screen and places the cursor in the upper left-hand corner of the screen. This corner is the point of reference for the symbol. This means that in symbol mode (when the symbol is later retrieved and integrated with a picture) the cursor will lie at the upper left-hand corner of the symbol.

Operations in symbol creation are similar to those in Draw Mode, although the repertoire of commands is slightly smaller. Specifically, there is no access to alphanumeric mode, fill mode, and symbol mode. The ARTIST program has to store a great deal of information as it creates and saves a symbol, and incorporating characters, filled areas, or other

[1] Programmers should note that the pictures are stored in the files PICFILxx.DAT, one picture per file. If no picture files are present when the ARTIST program is transferred to a new computer, ARTIST will create each file as it is needed, store a blank picture in it, and name the picture "*** Vacant."

symbols into a symbol could result in highly inefficient operation of the program. (I.e., the file space used and the time spent by the program in handling symbols could impose restraints that made the program inconvenient for the user.

The other commands of Draw Mode -- including cursor movement and backup mode -- are available for use during symbol definition, as shown in exhibit 7-1. The carriage return key causes the screen to switch between the help screen and the graphics screen.

Upon exiting the drawing phase (through the "Q" or ESC key), ARTIST asks if the symbol is to be saved and, if so, it also asks for a name to apply to the symbol. This is the name that ARTIST will show to the user in symbol mode when presenting a menu of available symbols from which the user can select. After the name is supplied, ARTIST stores the symbol's description in a file for later reference.[1]

7.3 SYMBOL DELETION

Symbols need not be deleted from the symbol file. Choosing the Delete Symbol item from the Main Menu causes ARTIST to display a menu of available symbols and to prompts for the number of a symbol to delete. For example:

```
Symbols available for deletion:

     0)  None
     1)  Motor rifle division
     2)  Tank division
     3)  Town

Enter symbol number (0-3)
```

[1] Programmers should note that this file, SYMBOLS.DAT, should be present in the same subdirectory as the ARTIST program. The file is of variable length and can hold any number of symbols within practical constraints of the capacity of the disk.

EXHIBIT 7-1: HELP SCREEN FOR SYMBOL DEFINITION

```
Symbol definition options:

     B - backup mode       U - pen up
     D - pen down          V - alter video default values


Pen Colors:

     E - erase (background color)
     C - cyan                 M - magenta     W - white
     G - green                R - red         Y - yellow


Cursor Movement:

     7 - ne                   8 - n           9 - nw
     4 - w                    5 -             6 - e
     1 - sw                   2 - s           3 - se


Other:

     Q, <esc> - exit drawing mode
     <CR>     - toggle between draw mode menu and graphic
                screen
```